

# Webdesign 3 – čeština a CSS

V minulém dílu seriálu jsme se již dotkli kódování textu. Nyní se na text a češtinu na internetu zaměříme ještě podrobněji. Pohlédneme především na styly textu a odstavců.

Dobrou zprávou hned na úvod našeho seriálu je, že na stránkách Amos Softwaru se objevila česká verze manuálu ke GoLive 5, 6 MB PDF soubor si můžete stáhnout na adrese: [www.amsoft.cz/produkty/adobe/golive5/downloademo.html](http://www.amsoft.cz/produkty/adobe/golive5/downloademo.html).

Jak jsme si ukázali minule, zaplnit internetové stránky textem není problém. Práce s ním je velmi podobná práci v jakémkoliv textovém editoru. O něco komplikovanější je jeho formátování...

## Kódování

Pro různé světové jazyky existují různá kódování. V anglo-amerických zemích se nejčastěji používá kódování ISO 8859-1 nebo nově UTF-8 (Unicode).

Pro země střední Evropy (Central European) byla Mezinárodní standardizační organizací vytvořena norma ISO 8859-2. Tato norma je doporučována i pro české texty na internetu.

Ti, kdož pracují s internetem delší dobu, si jistě pamatují, jak komplikovaně byla čeština webu řešena ještě před pár lety. Uživatel musel sám zvolit v jaké znakové sadě chce stránky prohlížet. To již dnes není nutné, stačí do hlavičky HTML kódu vložit tag, který prohlížeči sdělí, v jakém kódování byly stránky vytvořeny. Prohlížeč pak stránky sám správně překládá a zobrazí. Tento tag vložíte v GoLive do stránky tím, že v menu zvolíte: File > Document Encoding > Central European (ISO-8859-2).

Minule jsme doporučovali vložit kódování Windows 1250. Jaký je v obou kódováních rozdíl? Obě tato kódování fungují s češtinou bezproblémově na PC i na Macintoshi.

U Unixových systémů je jednoznačně preferováno kódování ISO 8859-2. Vypadá to tedy, že je lepší kódování ISO. Jenže z typografického hlediska je lepší kódování Windows 1250. Obsahuje totiž znaky jako N pomlčka (–), M pomlčka (—), dolní a horní uvozovky („“), ochranné značky (©, ®, ™) a další. Kódování ISO 8859-2 má místo pomlčky pouze minus (-), místo uvozovek palce ("), a copyright můžete napsat jen jako C v závorce (c).

Přehledné tabulky se všemi znaky obou kódování najdete například na <http://euro.typo.cz/typesetting/encodings.pdf>.

V praxi může být limitující to, že v rámci jedné HTML stránky nemůžete současně použít různá kódování. Tím je sazba například češtiny s ruštinou na jedné HTML stránce neuskutečnitelná.

Pro typografy bude v budoucnu záchranou přechod na Unicode – univerzální kódování, které do sebe pojme všechna světová kódování bez nutnosti konverze. Více o Unicode se můžete dozvědět na [www.unicode.org](http://www.unicode.org). Unicode je podporován na PC od Windows 98 a na Macintoshi od MacOS X. Na webu však ještě nestihl zdomácnět.

## Písmo

Mezi lidmi existuje mínění, že na internetu se dají použít pouze písma Arial (Helvetica) a Times. Není tomu tak docela. Písma můžete v HTML definovat dvěma způsoby.

### Generická písma

První z nich je generický název písma. Možné varianty jsou: proportional (default), sans-serif, serif, monospace, cursive a fantasy. Použijete-li v HTML generickou defi-

nici písma, budou uživatelé texty zobrazeny v písmech, která si pro tyto skupiny vybral. Uživatelé však většinou nemění výchozí nastavení, stránky se jim tedy zobrazí v písmech Times pro proportional a serif; Arial (resp. Helvetica) pro sans-serif a Courier pro Monospace. Některé prohlížeče dovolí nastavit pouze dva typy: neproporcionální písmo (Courier) a proporcionální písmo (Times). Generická písma se používají většinou jako „poslední záchrana“ nemá-li uživatel žádné z konkrétních písem.

### Konkrétní písma

HTML umožňuje použít jakékoliv písmo, které má čtenář stránek nainstalováno ve svém počítači. Nejčastěji se tedy na webu používají písma, která jsou součástí instalace operačního systému. Proto je nezbytné při specifikaci písem definovat vždy písma pro všechny operační systémy, ze kterých se uživatelé na vytvářené stránky připojují. U definovaného písma vždy záleží na přesném zápisu jeho jména (například Arial ≠ Ariel).

## Styly

V HTML lze definovat písma pro každý text či stránku zvlášť, ale vzhledem k tomu, že předpokládáme, že bude celý web vytvořen ze stejných písem, je nejlepší nadefinovat pro stránky styly (označují se jako Cascading Style Sheets).

### Interní kaskádové styly

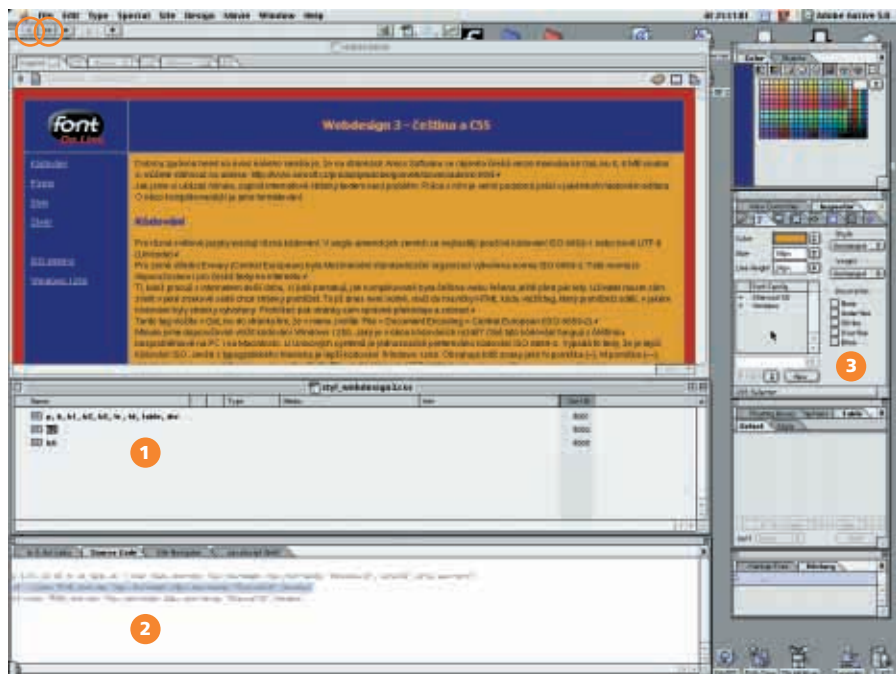
Kód specifikující interní styly se vkládá přímo do hlavičky HTML dokumentu a určuje písma pouze pro HTML dokument, v němž je obsažen. V dalším textu se interními styly zabývat nebudeme, i když v praxi mají své reálné opodstatnění.

### Externí kaskádové styly

Externí styly se ukládají do zvláštního souboru s příponou CSS. Nový soubor se styly založíte v GoLive následovně: File > New Special > Style Sheet Document. Po té se objeví prázdné okno s názvem „untitled.css“.

Podobně jako ve zlomových programech můžete definovat buď odstavcové styly nebo textové styly. Když je nové okno stylů aktivní, objeví se v kontextové liště (vlevo nahoře) znaky „“, „<>“ a „#“. Znakem

- Od minulých ukázek se toho na našich stránkách příliš nezměnilo. Veškerá odlišnost je v použitém písmu.
1. Okno, ve kterém GoLive uspořádává kaskádové styly.
  2. Zdrojový kód – těchto pár řádků může velmi dramaticky a bezbolestně ovlivnit vzhled celého webu, který jste vyrábíte.
  3. Inspector pomáhá při nastavování jednotlivých parametrů. Obsahuje několik záložek, kterými můžete ovlivňovat vzhled a zalomení a hypertextové odkazy v běžném HTML textu. (Dalšími parametry CSS se budeme zabývat příště.)



tečka se definují textové styly a znakem menší/větší se definují odstavcové styly. Znak „#“ se používá pro číselnou definici písma (užívá se především u dynamicky generovaných stránek), my se jím zabývat nebudeme.

### **Odstavcové styly (<p>, <b>, tag)**

Od minule jste si jistě za domácí úkol vyzkoušeli jak pracují v HTML předdefinované styly (tagy <p>, <b>, <h1>, ...). Vtip kaskádových stylů spočívá v tom, že s jejich pomocí můžete předdefinovat chování běžných stylů HTML (ikonka „<>“ značí tag). Stiskneme tedy ikonku „<>“ a zkusíme předdefinovat vizuální podobu například nadpisu o velikosti 2 (<h2>). Po stisknutí ikonky se nám v dosud prázdném okně objeví „<> element“. Přepíšeme slovo element na „h2“ a pustíme se v paletce Inspector do definování stylu. Z této paletky zvolíme záložku označenou písmenem „F“. Zde změním barvu na tmavě žlutou, velikost písma na 18 px a proklad na 24 px. Pixly udávají absolutní velikost písma v obrazovkových bodech. Jednotek, ve kterých se zadává velikost písma, je k dispozici celá řada. V zásadě se jedná o velikosti absolutní (například použité pixly) nebo relativní (např. 100 % nebo označení „medium“). Až budete se styly experimentovat, určitě si otevřete okno zdrojového kódu (Source Code) a sledujte v něm, jak GoLive za vás kód píše.

### **Typ písma**

Typ písma se definuje v okénku Font Family. Stiskem tlačítka „New“ přidáme nové písmo. Po té můžeme z menu vybrat požadovaná písma nebo napsat jejich přesné názvy. Častěji budeme asi názvy psát, protože nikdy nebudeme mít v systému k dispozici písma pro všechny používané systémy.

Postupně začneme plnit okénko Font Family následujícími písmi: Charcoal CE, Verdana, Helvetica CE, Arial CE, Arial, sans-serif. Browser použije vždy první písmo ze seznamu, které má k dispozici.

Chování například MS Exploreru 5 pro Macintosh bude následující: uvidí písmo Charcoal CE, zjistí, že ho má k dispozici (písmo je součástí českého systému MacOS) a zobrazí v něm všechny texty vyznačené jako nadpis 2 (<h2>).

Přijde-li na stránku s takto definovanými styly například uživatel Exploreru 4.5 pro Windows 98, systém bude postupovat následovně: vezme písmo Charcoal CE, zjistí, že ho nemá k dispozici; vezme další písmo – Verdanu, zjistí, že je k dispozici a zobrazí v něm všechny titulky velikosti 2.

Nezanedbatelné jsou také přípony „CE“. Jsou důležité pro počítače, které nepodporují Unicode (Win 3.x, Win 95, Mac OS do verze 9.x). Říkají systému, že mají použít písmo se střeoevropskou znakovou sadou. Unicodové systémy tuto informaci

nepotřebují, protože například písmo s názvem „Arial“ obsahuje i znaky východoevropské.

Vezmeme-li v úvahu například Windows 3.11 s Explorerem 4, jejich „uvažování“ by bylo následující: nerozeznaly by písma Charcoal CE, Verdana, ani Helvetica CE (nejsou součástí Win 3.11). Použily by až písmo Arial CE. Kdyby však například bylo písmo „Arial“ před písmem „Arial CE“, použily by Win 3.11 písmo Arial a čeština by se neobrazila správně... Tato chyba často zapříčiňuje chybné zobrazení češtiny i na počítačích Macintosh. Zapamatujte si, že je nezbytné vždy nejprve uvádět názvy písem s příponou CE (EE) a po té teprve bez ní.

Pro všechny případy, kdyby si někdo vymazal všechna definovaná písma ze systému, je v definici písma také sans-serif, který zajistí použití bezserifového písma.

V úplně krajním případě, kdyby uživatel neměl ani žádné bezserifové písmo, použije prohlížeč písmo výchozí (nejčastěji Times).

Nyní máme nadefinovány základní parametry pro nadpis o velikosti 2 (h2), je tedy čas styl uložit. Pojmenujeme ho například „styl\_webdesign3.css“. (V minulém díle jsme nezdůraznili, že je bezpečnější pojmenovávat soubory malými písmeny bez mezer a dalších speciálních znaků.)

### **Styly textu („.“, class)**

Definice stylu textu je obdobná jako u stylu odstavců, odlišné je jen použití a dědičnost. Vyzkoušejte si aplikovat styly textu a styly odstavců současně, pochopíte tak nejlépe dědičnost, resp. kaskádovost.

### **Použití CSS na stránce**

Po uložení stylu jej můžete v GoLive aplikovat na stránku pouhým přetažením souboru CSS do záhlaví stránky. Odstavcové styly vytvořené pro standardní předdefinované HTML tagy se ihned automaticky aplikují na všechny texty ve stránce.

### **Proč se styly nazývají kaskádové?**

Kaskádové styly (a HTML také) mají definovanou dědičnost parametrů. Nadefinujete-li tedy určitému tagu nějaké písmo a dalším tagem ho nezměníte, bude i text obsažený v tomto tagu zobrazen stejným písmem. Dědičnost nejlépe pochopíte, když si z internetu stáhnete námi vytvořený styl a zkusíte s nimi experimentovat. Tento článek naleznete v HTML na adrese [www.kafka.cz/font/webdesign3](http://www.kafka.cz/font/webdesign3) (přesnou cestu ke stylu neleznete v hlavičce kódu).

### **Závěr**

Jak jste jistě zjistili, není definice stylů záležitostí pár minut. Vytvoření kvalitních stylů zabere spoustu času, znalostí o operačních systémech (písmech v nich předinstalovaných) a testování. Smutné je také to, že jednotlivé browsery, byť na stejné platformě, se liší v zobrazování jednotlivých stylů...

-JT-